

# 05 - Checking Normality, Transformations to Near Normality

Junvie Pailden

SIUE, F2017, Stat 589

September 04, 2017

## Investigating Univariate Normality

- Could check each of the  $p$  variables for normality. Should not be the sole approach because variables are correlated and normality of individual variables does not guarantee multivariate normality.
- However, multivariate normality implies individual normality. Thus, if one of the variables is not univariate normal, then the vector is not multivariate normal.
- One check Q-Q (Quantile-Quantile) plot or normal probability plot for each variable. If normal, then the Q-Q plot is a straight line - subjective.

## Constructing Q-Q plots

- Order the original observations to get  $x_{(1)}, x_{(2)}, \dots, x_{(n)}$  and their corresponding probability values  $(1 - \frac{1}{2})/n, (2 - \frac{1}{2})/n, \dots, (n - \frac{1}{2})/n$ ;
- Calculate the standard normal quantiles  $q_{(1)}, q_{(2)}, \dots, q_{(n)}$ ; and
- Plot the pairs of observations  $(q_{(1)}, x_{(1)}), (q_{(2)}, x_{(2)}), \dots, (q_{(n)}, x_{(n)})$ , and examine the “straightness” of the outcome.

## Investigating Bivariate Normality

- Check the scatter plot of each pair of the  $p$  variables.
- The points on each scatter plot should form approximately an ellipse since the contours of Bivariate Normal Distributions are ellipses.
- The set of generalized distances from each point to the center of the points is chi-square - Check using a chi-square plot.

## Example: Reaven and Miller (1979)

Reaven and Miller measured five variables in a comparison of normal patients and diabetics. We use partial data for normal patients only. The three variables of major interest were

- $X_1$  = glucose intolerance,
- $X_2$  = insulin response to oral glucose,
- $X_3$  = insulin resistance.

The two additional variables of minor interest were

- \*  $Y_1$  = relative weight,
- \*  $Y_2$  = fasting plasma glucose.

```
# load data set
```

```
patients <- read.csv("patients.csv", header = TRUE)  
str(patients) # structure
```

```
# 'data.frame': 46 obs. of 5 variables:
```

```
# $ WEIGHT : num 0.81 0.95 0.94 1.04 1 0.76 0.91 1.1 0.99
```

```
# $ FASTING: int 80 97 105 90 90 86 100 85 97 97 ...
```

```
# $ GLUCOSE: int 356 289 319 356 323 381 350 301 379 296
```

```
# $ INSULIN: int 124 117 143 199 240 157 221 186 142 131
```

```
# $ RESIST : int 55 76 105 108 143 165 119 105 98 94 ...
```

## Descriptive Summary Statistics

```
data.frame(Mean = colMeans(patients),  
           Median = apply(patients, 2, median),  
           Variance = apply(patients, 2, var))
```

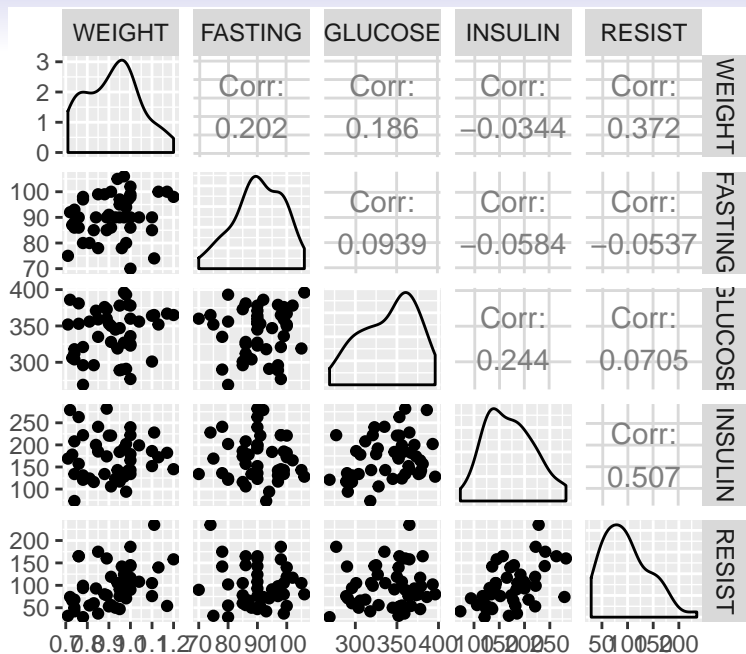
#		Mean	Median	Variance
#	WEIGHT	0.92	0.94	1.6e-02
#	FASTING	90.41	90.00	7.1e+01
#	GLUCOSE	340.83	351.00	1.1e+03
#	INSULIN	171.37	170.50	2.4e+03
#	RESIST	97.78	92.00	2.1e+03

Check each variable, scatterplots, density estimates,  
QQ-Plots

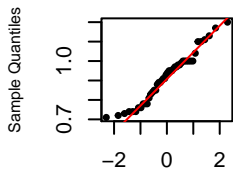
```
GGally::ggpairs(patients) # ggpairs() in package GGally

par(mfcol = c(2,3)) # 2x3 panel
for (p in 1:5){ # for loop qqplot for each variable
  qqnorm(patients[, p], # qqplot - normal
    main = paste(colnames(patients)[p]),
    cex = 0.8, cex.lab = 0.7, pch = 20)
  # use variable name as title
  qqline(patients[, p], col = "red") # draw line
}
```

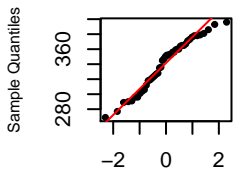




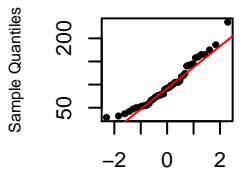
## WEIGHT



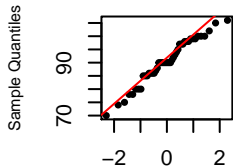
## GLUCOSE



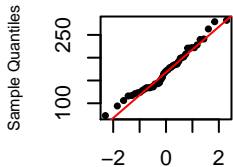
## RESIST



## FASTING



## INSULIN



## Univariate Normality Test

- The Kolmogorov-Smirnov (KS) test is used to decide if a sample comes from a population with a specific distribution.
  - test statistic does not depend on the underlying cumulative distribution function being tested (non-parametric)
  - exact test, not based on large-sample approximation
  - applies to continuous variables
  - tends to be more sensitive near the center of the distribution than at the tails.
- Null Hypothesis: The data follow a specified distribution

$$D = \max_{1 \leq i \leq n} \left( F(Y_i) - \frac{i-1}{n}, \frac{i}{n} - F(Y_i) \right)$$

-  $F$  is the theoretical CDF being tested (normal CDF if testing for normality)

```
x1 <- rexp(100) # sample from std. exponential dist'n  
ks.test(x1, "pnorm")
```

```
#  
# One-sample Kolmogorov-Smirnov test  
#  
# data: x1  
# D = 0.5, p-value <2e-16  
# alternative hypothesis: two-sided
```

```
x2 <- rt(100, df = 3) # sample from t-dist with df = 3  
ks.test(x2, "pnorm") # data has same center as normal
```

```
#  
# One-sample Kolmogorov-Smirnov test  
#  
# data: x2  
# D = 0.1, p-value = 0.2  
# alternative hypothesis: two-sided
```

## Anderson-Darling Test

Anderson-Darling test is a KS-test variant which is sensitive to tail distribution variation/changes.

```
# if not installed, run install.packages("nortest") in your  
nortest::ad.test(x2) # need nortest package
```

```
#  
#   Anderson-Darling normality test  
#  
# data:  x2  
# A = 2, p-value = 7e-05
```

## Reaven and Miller (1979) Data

```
tstat <- pval <- rep(NA, 5)
for(p in 1:5){
  test.out <- nortest::ad.test(patients[, p])
  tstat[p] <- test.out$statistic
  pval[p] <- test.out$p.value
}
data.frame(variable = colnames(patients),
            AD.statistic = tstat, p.value = pval)
```

#	variable	AD.statistic	p.value
# 1	WEIGHT	0.52	0.179
# 2	FASTING	0.53	0.167
# 3	GLUCOSE	0.68	0.069
# 4	INSULIN	0.46	0.251
# 5	RESIST	0.68	0.070

## Assessing Multivariate Normality - Chi-Square Plot

1. Compute the generalized squared (Mahalanobis) distance from each  $\mathbf{X}_i$  to  $\bar{\mathbf{X}}$  given by

$$D_i^2 = (\mathbf{X}_i - \bar{\mathbf{X}})' \mathbf{S}^{-1} (\mathbf{X}_i - \bar{\mathbf{X}})$$

2. List  $D_i^2$  from low to high. If  $\mathbf{X}_i$  are multivariate normal, then  $D_i^2$  has a Chi-Squared distribution.)
3. Form a  $Q - Q$  plot based on the Chi-Squared distribution.  
If the line to a straight line, then it is reasonable to assume multivariate normal.

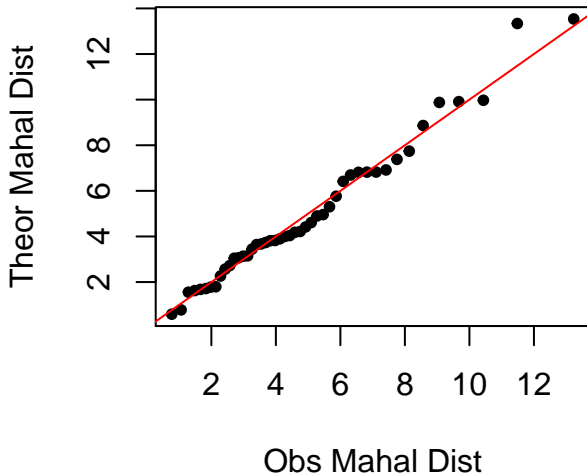
## Chi-Square plots

```
ChiSq.plot <- function(data, main = "Chi-Square Plot"){  
  # function for drawing chi-square plots  
  x <- as.matrix(data)  
  n <- nrow(data)  
  xbar <- colMeans(data) # col means  
  S <- var(data) # covariance matrix  
  di2 <- rep(0,n) # storage for MD  
  for (i in 1:n){      # MD distance for each observation  
    di2[i] <- t(x[i,]-xbar) %*% solve(S) %*% (x[i,]-xbar)}  
  CP.dat <- data.frame(expvals = qchisq((1:n)/(n+1),5) ,  
    obsvals = sort(di2))  
  plot(CP.dat, pch =20, main = main, # plot the points  
    xlab = "Obs Mahal Dist", ylab = "Theor Mahal Dist")  
  lines(c(0,20), c(0,20), col="red")  
}
```



```
ChiSq.plot(patients)
```

## Chi-Square Plot



## Multivariate Normal Goodness of Fit Test - Energy Test

The E-test of multivariate normality was proposed and implemented by Szekely and Rizzo (2005). The E-test of multivariate normality is implemented by parametric bootstrap with  $R$  replicates.

```
# mvnrm.etest() is in energy package  
# R is number of bootstrap replicates  
energy::mvnrm.etest(patients, R = 199)
```

```
#  
#   Energy test of multivariate normality: estimated param  
#  
# data:  x, sample size 46, dimension 5, replicates 199  
# E-statistic = 1, p-value = 0.5
```

## Radiation Levels Data

- The quality control department of a manufacturer of microwave ovens is required by the federal government to monitor the amount of radiation emitted when the doors of the ovens are closed.
- Observations of the radiation emitted through closed and open doors of  $n = 42$  randomly selected ovens were made.

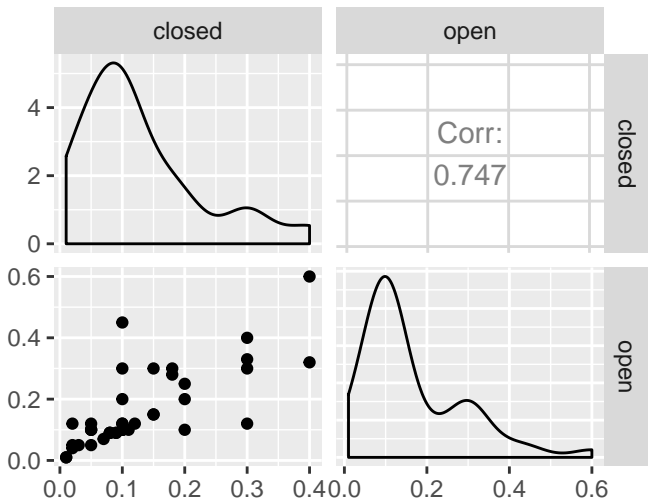
```
radiation <- read.table("radiation.txt", header = F)
colnames(radiation) <- c("closed", "open")
moments::skewness(radiation)
```

```
# closed    open
#      1.2    1.5
```

```
# both variables are skewed to the right
```

## Radiation Data Scatterplot

```
GGally::ggpairs(radiation)
```



## Radiation Data, MVN Goodness of Fit Test

```
energy::mvnrm.etest(radiation, R = 500)
```

```
#  
#   Energy test of multivariate normality: estimated parameters  
#  
# data:  x, sample size 42, dimension 2, replicates 500  
# E-statistic = 4, p-value <2e-16
```

## Box-Cox Transformations

- A Box-Cox transformation of a variable  $x$  is defined as

$$x^{(\lambda)} = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \ln x & \lambda = 0 \end{cases}$$

which is continuous in  $\lambda$  for  $x > 0$ .

- These transformations are data based in the sense that it is only the appearance of the data that influences the choice of an appropriate transformation index by  $\lambda$ .
- Given observations  $x_1, \dots, x_n$ , the Box-Cox solution for the choice of an appropriate power  $\lambda$  is the solution that maximizes the expression

$$\ell(\lambda) = -\frac{n}{2} \ln \left[ \frac{1}{n} \sum_{j=1}^n (x_j^{(\lambda)} - \bar{x}^{(\lambda)})^2 \right] + (\lambda - 1) \sum_{j=1}^n \ln x_j$$

## Box-Cox Transformations in R

```
library(car)
# compute appropriate power transformation
lam <- powerTransform(radiation)
summary(lam)
```

```
# bcPower Transformations to Multinormality
#           Est Power Rounded Pwr Wald Lwr bnd Wald Upr Bnd
# closed      0.16           0      -0.105           0.43
# open        0.15           0      -0.071           0.37
#
# Likelihood ratio tests about transformation parameters
#           LRT df      pval
# LR test, lambda = (0 0)  2.3  2 3.1e-01
# LR test, lambda = (1 1) 51.1  2 7.9e-12
```

## Transformed the Radiation Data

```
coef(lam)
```

```
# closed    open  
#   0.16    0.15
```

```
radiation.transformed <- bcPower(with(radiation,  
                                     cbind(closed, open)), coef(lam))
```

```
moments::skewness(radiation.transformed) # skewness
```

```
# closed^0.16    open^0.15  
#      -0.29      -0.30
```



## Transformed Radiation Data Scatterplot

```
GGally::ggpairs(data.frame(radiation.transformed))
```

